

# UML 2.5+

## Introducción



# ¿Qué es UML?

El lenguaje unificado de modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el Object Management Group (OMG).

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, **no es programación**, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que programación estructurada es una forma de programar como lo es la orientación a objetos, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML solo para lenguajes orientados a objetos.

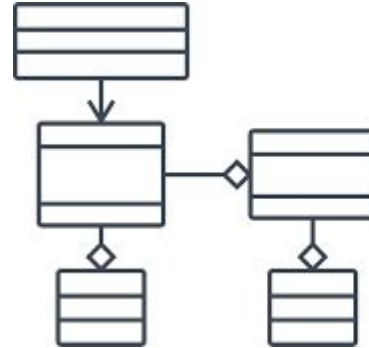
*Ángel González M.*

# ¿Por qué UML?

Es el lenguaje estándar más usado para modelar software.

En tu trabajo diario, vas a necesitar leer diagramas escritos por otras personas.

Un proyecto puede ser muy simple, o puede estar constituido por cientos o miles de piezas de código y hardware.

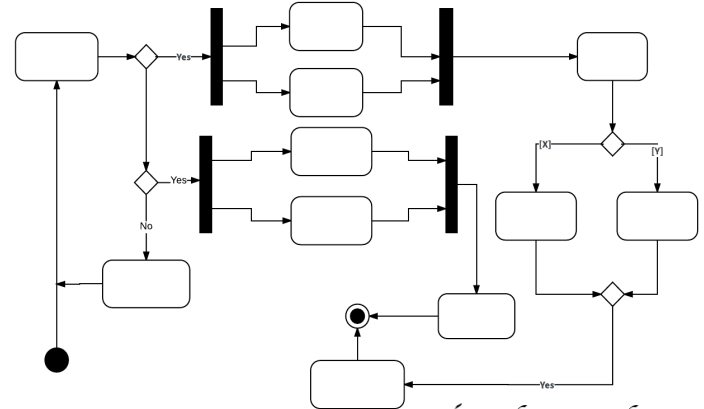


# ¿Por qué UML?

Para controlar la complejidad, permitiendo enfocarme en los aspectos importantes del diseño

UML nos permite comunicar complicados aspectos técnicos a los demás desarrolladores, analistas, diseñadores, testadores y clientes

Esto aumentará la productividad, permitiéndonos centrarnos en construir sistemas de más calidad en menos tiempo.



Ángel González M.

# Tipos de diagramas en UML 2.5

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.

Diagrama de Casos de Uso



Diagrama de Clases

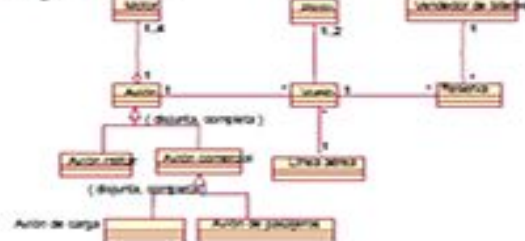


Diagrama de Secuencia



Diagrama de Colaboración

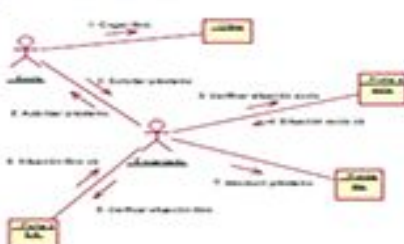


Diagrama de Estados

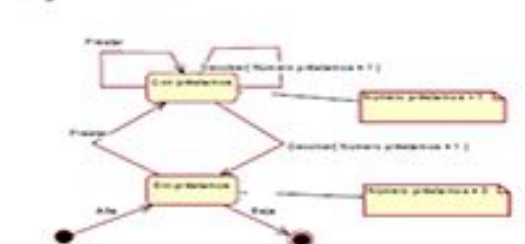


Diagrama de Actividad

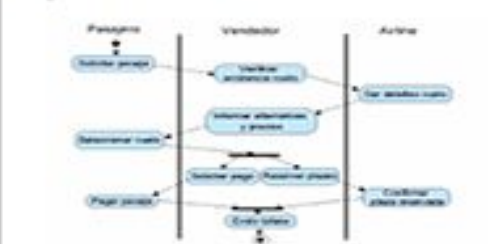


Diagrama de Objetos



Diagrama de Componentes

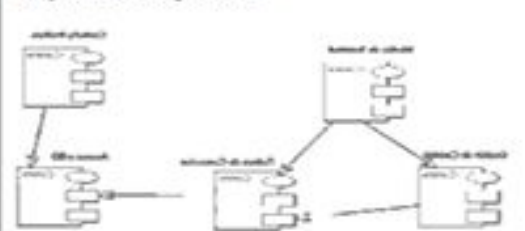
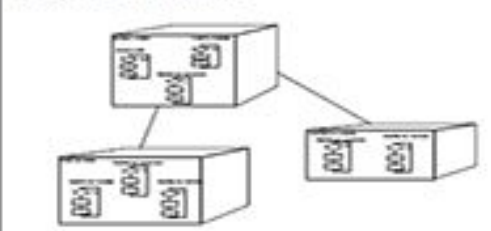


Diagrama de Distribución



# Estructurales

Muestran la estructura estática de los objetos en un sistema.

- **Diagrama de clases** Los diagramas de clase son, sin duda, el tipo de diagrama UML más utilizado. Es el bloque de construcción principal de cualquier solución orientada a objetos. **Muestra las clases en un sistema, atributos y operaciones de cada clase y la relación entre cada clase.** En la mayoría de las herramientas de modelado, una clase tiene tres partes, nombre en la parte superior, atributos en el centro y operaciones o métodos en la parte inferior. En sistemas grandes con muchas clases relacionadas, las clases se agrupan para crear diagramas de clases. Las Diferentes relaciones entre las clases se muestran por diferentes tipos de flechas.
- **Diagrama de componentes** Un diagrama de componentes muestra la relación estructural de los componentes de un sistema de software. Estos se utilizan principalmente **cuando se trabaja con sistemas complejos que tienen muchos componentes.** Los componentes se comunican entre sí mediante interfaces. Las interfaces se enlazan mediante conectores.
- **Diagrama de despliegue** Un diagrama de despliegue muestra el hardware de su sistema y el software de ese hardware. Los diagramas de implementación son **útiles cuando la solución de software se despliega en varios equipos**, cada uno con una configuración única.
- **Diagrama de objetos** Los diagramas de objetos, a veces denominados diagramas de instancia, son muy similares a los diagramas de clases. Al igual que los diagramas de clases, también **muestran la relación entre los objetos, pero usan ejemplos del mundo real.** Se utilizan para mostrar cómo se verá un sistema en un momento dado. Debido a que hay datos disponibles en los objetos, a menudo se utilizan para explicar relaciones complejas entre objetos.
- **Diagrama de paquetes** Como su nombre indica, un diagrama de paquetes **muestra las dependencias** entre diferentes paquetes de un sistema.
- **Diagrama de perfiles** El diagrama de perfil es un nuevo tipo de diagrama introducido en UML 2. Este es un tipo de diagrama que se utiliza muy raramente en cualquier especificación.
- **Diagrama de estructura compuesta** Los diagramas de estructura compuesta se utilizan para **mostrar la estructura interna de una clase.**

# De comportamiento

Muestran el comportamiento dinámico de los objetos en el sistema.

- **Diagrama de actividades** Los diagramas de actividad representan los **flujos de trabajo de forma gráfica**. Pueden utilizarse para describir el flujo de trabajo empresarial o el flujo de trabajo operativo de cualquier componente de un sistema. A veces, los diagramas de actividad se utilizan como una alternativa a los diagramas de máquina del estado.
- **Diagrama de casos de uso** Como el tipo de diagrama de diagramas UML más conocido, los diagramas de casos de uso **ofrecen una visión general de los actores involucrados en un sistema**, las diferentes funciones que necesitan esos actores y cómo interactúan estas diferentes funciones. Es un gran punto de partida para cualquier discusión del proyecto, ya que se pueden identificar fácilmente los principales actores involucrados y los principales procesos del sistema.
- **Diagrama de máquina de estados** Los diagramas de máquina de estado son similares a los diagramas de actividad, aunque las anotaciones y el uso cambian un poco. En algún momento se conocen como diagramas de estados o diagramas de diagramas de estado también. Estos son muy **útiles para describir el comportamiento de los objetos que actúan de manera diferente de acuerdo con el estado** en que se encuentran en el momento.



# De interacción

- **Diagrama global de interacciones** Los diagramas generales o globales de interacción son muy similares a los diagramas de actividad. Mientras que los diagramas de actividad muestran una secuencia de procesos, los diagramas de interacción **muestran una secuencia de diagramas de interacción**. En términos simples, pueden llamarse una colección de diagramas de interacción y el orden en que suceden. Como se mencionó anteriormente, hay siete tipos de diagramas de interacción, por lo que cualquiera de ellos puede ser un nodo en un diagrama de vista general de interacción.
- **Diagrama de comunicación** El diagrama de comunicación se llamó diagrama de colaboración en UML 1. Es similar a los diagramas de secuencia, pero **el foco está en los mensajes pasados entre objetos**.
- **Diagrama de secuencia** Los diagramas de secuencia en UML muestran **cómo los objetos interactúan entre sí y el orden en que se producen esas interacciones**. Es importante tener en cuenta que muestran las interacciones para un escenario en particular. Los procesos se representan verticalmente y las interacciones se muestran como flechas.
- **Diagrama de tiempos** Los diagramas de sincronización son muy similares a los diagramas de secuencia. **Representan el comportamiento de los objetos en un marco de tiempo dado**. Si es solo un objeto, el diagrama es directo, pero si hay más de un objeto involucrado, también se pueden usar para mostrar interacciones de objetos durante ese período de tiempo.

# ¿Qué diagramas vamos a aprender?

- Diagramas de casos de uso
- Especificación de casos de uso
- Diagramas de actividad
- Diagramas de clases
- Diagramas de objetos
- Diagramas de paquetes
- Diagramas de secuencia
- Diagramas de transición de estados
- Otros